In the Claims

Please amend the claims as shown below.

1. (Canceled)

2. (Previously presented) The method of claim 33, wherein the design automation tool is used to implement hardware descriptor language designs on a programmable chip.

3. (Currently amended) The method of claim 33, wherein the plurality of submodules comprises a memory module and a Digital Signal Processor (DSP) core.

4. (Currently amended) The method of claim 33, ~~wherein~~ further comprising applying a plurality of instantiation constraints ~~are used~~ to select the plurality of submodules.

5. (Previously presented) The method of claim 33, wherein the design automation tool is a synthesis or a place and route tool.

6. (Canceled)

7. (Previously presented) The method of claim 33, further comprising identifying a plurality of inputs, wherein the inputs identified comprise the input pins of the module, one of the output lines of one of the parameterized submodules, and registers.

8. (Previously presented) The method of claim 7, further comprising identifying a plurality of outputs, wherein the outputs identified comprise the output pins of the module, one of the input lines of one of the parameterized submodules, and registers.

9. (Previously presented) The method of claim 8, further comprising classifying the inputs and outputs identified as clock lines, control lines, and data lines.

10-12. (Canceled)

13. (Currently amended) The method of claim 33, wherein parameterizing the plurality of submodules comprises defining interfaces, data widths, and the types of signals for one of the input lines and one of the output lines associated with one of the parameterized submodules.

14. (Previously presented) The method of claim 33, wherein the submodules comprise adders, phase lock loops, memory, and timers.

15. (Previously presented) The method of claim 9, wherein one of the test designs further comprises a clock structure for one of the outputs.

16. (Previously presented) The method of claim 15, wherein the clock structure includes a synchronous and or an asynchronous structure.

17. (Canceled)

18. (Previously presented) The computer system of claim 36, wherein the design automation tool is used to implement hardware descriptor language designs on a programmable chip.

19. (Previously presented) The computer system of claim 36, wherein the design automation tool is used to implement designs on an ASIC.

20. (Previously presented) The computer system of claim 36, wherein the design automation tool is an electronic design automation tool.

21. (Previously presented) The computer system of claim 36, wherein the design automation tool is a synthesis or a place and route tool.

22. (Canceled)

23. (Currently amended) The computer system of claim 36, wherein said processor is ~~configured~~ operable to generate [[the]] one of the ~~plurality of~~ test designs by identifying a

plurality of inputs, wherein the plurality of inputs comprises the input pins of the top level module, one of the output lines of one of the parameterized submodules, and registers.

24. (Currently amended) The computer system of claim 23, wherein said processor is ~~configured~~ operable to generate [[the]] one of the ~~plurality of~~ test designs by identifying a plurality of outputs, wherein the outputs identified comprise the output pins of the top level module, one of the input lines of one of the parameterized submodules, and registers.

25. (Canceled)

26. (Previously presented) The apparatus of claim 39, wherein the design automation tool is used to implement hardware descriptor language designs on a programmable chip.

27. (Previously presented) The apparatus claim 39, wherein the design automation tool is used to implement designs on an ASIC.

28. (Previously presented) The method of claim 35, further comprising selecting a plurality of submodules upon said determining that the predetermined number of the test designs is not generated.

29-32. (Canceled)

33. (Currently amended) A method ~~of generating a plurality of test designs associated with a design automation tool, the method~~ comprising:

generating a plurality of test designs associated with a design automation tool, wherein plurality of test designs comprises a module, wherein said generating the plurality of test designs comprises:

(a) instantiating an input/output (I/O) structure of [[a]] the module having input and output pins;

(b) applying a function to select a plurality of submodules from a design module library, wherein the plurality of submodules comprises input and output lines, wherein the plurality of submodules is located within the module;

(c) parameterizing the plurality of submodules from the design module library for interconnection with the module;

(d) interconnecting the lines of the plurality of parameterized submodules based on a selection of a particular type of interconnect from a plurality of interconnect types, wherein the selection of the particular type of interconnect is based on a probabilistic function; and

(f) connecting the plurality of parameterized submodules to the input and output pins of the module.

34. (Currently amended) The method of claim 33, wherein the plurality of interconnect types includes an interconnect having a mathematical expression, an interconnect having conditional logic, or a direct interconnect.

35. (Previously presented) The method of claim 33, further comprising:

determining whether a predetermined number of the test designs for testing the design automation tool has been generated; and

applying the plurality of test designs to test the design automation tool.

36. (Currently amended) A computer system ~~for generating a plurality of test designs associated with a design automation tool, the computer system~~ comprising:

a memory operable to hold information associated with a design module library; and

a processor coupled to memory, wherein said processor is operable to generate a plurality of test designs associated with a design automation tool, wherein the plurality of test designs comprises the module, said processor is ~~configured~~ operable to:

(a) instantiate an input/output (I/O) structure of [[a]] the module having input and output pins;

(b) apply a function to select a plurality of submodules from the design module library, wherein the plurality of submodules ~~have~~ [[s]] includes a plurality of input and output lines, wherein the plurality of submodules is located within the module;

(c) parameterizing the plurality of submodules from the design module library for interconnection with the module;

(d) interconnect the lines of the plurality of parameterized submodules based on a selection of a particular type of interconnect from a plurality of interconnect types, wherein the selection of the particular type of interconnect is based on a probabilistic function; and

(e) connect the plurality of parameterized submodules to the input and output pins of the module.

37. (Currently amended) The computer system of claim 36, wherein the plurality of interconnect types includes an interconnect having a mathematical expression, an interconnect having conditional logic, or a direct interconnect.

38. (Currently amended) The computer system of claim 36, wherein the processor is ~~configured~~ operable to determine whether a predetermined number of the test designs for testing the design automation tool has been generated, wherein the plurality of test designs are applied to test the design automation tool.

39. (Currently amended) An apparatus ~~for generating a plurality of test designs associated with a design automation tool, the apparatus~~ comprising:

storage means for storing data design module library; and

processing means for generating a plurality of test designs associated with a design automation tool, wherein the plurality of test designs comprises a module, said processing means:

(a) for instantiating an input/output (I/O) structure of [[a]] the module having input and output pins,

(b) for applying a function to select a plurality of submodules having input and output lines from the design module library, wherein the plurality of submodules is located within the module,

(c) for parameterizing the plurality of submodules from the design module library for interconnection with the module,

(d) for interconnecting the lines of the plurality of parameterized submodules based on a selection of a particular type of interconnect from a plurality of interconnect types, wherein the selection of the particular type of interconnect is based on a probabilistic function; and

(e) for connecting the plurality of parameterized submodules to the input and output pins of the module.

40. (Currently amended) The apparatus of claim 39, wherein said processing means is further:

[[(f)]] for determining whether a predetermined number of the test designs for testing the design automation tool has been generated, and

[[(g)]] for repeating selecting a plurality of submodules from the design module library upon determining that the predetermined number of test designs is not generated.

41. (Currently Amended) The apparatus of claim 39, wherein the processing means is further for interconnecting the lines of the plurality of parameterized submodules based on the selection of the particular type of interconnect from the plurality of interconnect types including an interconnect having a mathematical expression, an interconnect having conditional logic, or a direct interconnect.

42. (New) The method of claim 33, further comprising receiving the probabilistic function from a user, wherein the probabilistic function comprises a probability of selecting one of the submodules from the plurality of submodules.